IN THE CLAIMS:

This listing of the claims replaces all prior versions and listings of the claims in this application.

The text of all pending claims (including any withdrawn claims) is set forth below. Canceled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (Original), (Currently amended), (Canceled), (Withdrawn), (Previously presented), (New), and (Not entered).

Please AMEND claims 1-10, 12, and 17-26 and ADD new claim 28 in accordance with the following:

1. (Currently amended) An apparatus for reproducing audio video (AV) data using a markup document in an interactive mode selected by a user of the apparatus, comprising:

a buffer ~~which buffers~~ to buffer the markup document to enable the apparatus to reproduce the AV data in the interactive mode selected by the user; and

a buffer manager ~~which manages~~ to manage the buffer to preload the markup document and ~~outputs~~ output buffering state information of the buffer in response to a report signal, the buffering state information being used by the apparatus in reproducing the AV data in the interactive mode selected by the user.

2. (Currently amended) The apparatus of claim 1 further comprising a content decoder ~~which interprets~~ to interpret the markup document and ~~outputs~~ output the report signal;~~,~~

wherein the buffer manager informs the content decoder of the buffering state information of the buffer in response to the report signal.

3. (Currently amended) The apparatus of claim 2, wherein the content decoder generates the report signal using an application program interface (API).

4. (Currently amended) The apparatus of claim 3, wherein the API ~~serves to notify~~ notifies the content decoder of whether preloading of the markup document has succeeded or failed, or whether the markup document is still being loaded.

5. (Currently amended) The apparatus of claim 4, wherein the API returns a value of 0 ~~where~~ if the preloading of the markup document has succeeded, returns a value of 1 ~~where~~ if the preloading of the markup document has failed, and returns a value of 2 ~~where~~ if the markup document is still being loaded.

6. (Currently amended) The apparatus of claim 2, wherein the content decoder generates the report signal using an API~~, which includes at least one of~~ comprising a file path ~~and~~ and/or an attribute of the markup document as a parameter.

7. (Currently amended) The apparatus of claim 2, wherein the content decoder generates the report signal using an [obj].isCached(URL, resType) API, where ~~the~~ URL is a parameter indicating a file path of the markup document, and ~~the~~ resType is a parameter indicating an attribute of the markup document.

8. (Currently amended) The apparatus of claim 2, wherein the buffer manager informs the content decoder of a buffering state of the markup document ~~utilizing~~ using an API.

9. (Currently amended) The apparatus of claim 1, further comprising a content decoder ~~which interprets~~ to interpret the markup document;~~,~~

wherein the buffer manager transfers the markup document from the buffer to the content decoder in response to a reproduce signal.

10. (Currently amended) The apparatus of claim ~~1~~ 9, ~~further comprising a content decoder which interprets the markup document,~~ wherein the content decoder outputs a release signal to the buffer manager indicating that the markup document ~~therein brought~~ that was transferred from the buffer~~,~~ to the content decoder in response to ~~a~~ the reproduce signal~~,~~ is not in use.

11. (Original) The apparatus of claim 10, wherein the content decoder outputs the release signal to the buffer manager in response to the markup document no longer being displayed in a screen of a display device.

12. (Currently amended) The apparatus of claim 1 further comprising a content decoder ~~which interprets~~ to interpret the markup document;~~,~~

wherein the buffer manager deletes the markup document from the buffer in response to a discard signal output from the content decoder.

13. (Original) The apparatus of claim 12, wherein the content decoder generates the discard signal using a discard API.

14. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a progressNameOfFile API to determine a file name of the markup document currently being preloaded.

15. (Original) The apparatus of claim 2, wherein the content decoder generates the report signal using a progressLengthOfFile API to determine how much of the markup document currently being preloaded has been preloaded.

16. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a remainLengthOfFile API to determine how much of the markup document currently being preloaded is yet to be preloaded.

17. (Currently amended) The apparatus of claim 2, wherein the content decoder generates the report signal using a totalLoadingSize API to determine a total ~~load~~ loading size of the markup document to be preloaded.

18. (Currently amended) The apparatus of claim 2, wherein the content decoder generates the report signal using a remainLoadingSize API to determine how much of a total ~~load~~ loading size of the markup document ~~is~~ has yet to be preloaded.

19. (Currently amended) An apparatus for controlling a buffer ~~which buffers~~ to buffer a markup document to reproduce audio video (AV) data in an interactive mode selected by a user of the apparatus, the apparatus comprising:

a buffer manager ~~which manages~~ to manage the buffer to preload the markup document to enable the apparatus to reproduce the AV data in the interactive mode selected by the user, and ~~outputs~~ output information of the buffer ~~including~~ comprising buffering information of the markup document;~~,~~

wherein the buffering information ~~includes~~ is used by the apparatus in reproducing the AV data in the interactive mode selected by the user and comprises:

information indicating that preloading of the markup document has succeeded;

information indicating that the preloading of the markup document has failed; and

information indicating that the preloading of the markup document is still ~~be~~ being conducted.

20. (Currently amended) The apparatus of claim 19, wherein the buffer manager outputs the information of the buffer using an application program interface (API).

21. (Currently amended) The apparatus of claim 19, wherein the information of the buffer further ~~includes~~ comprises information indicating whether a command to preload the markup document has been successfully received.

22. (Currently amended) The apparatus of claim 19, wherein the information of the buffer further ~~includes~~ comprises information indicating whether preloading of the markup document ~~is~~ has been completed.

23. (Currently amended) An apparatus for recording and/or reproducing audio video (AV) data using a markup document in an interactive mode selected by a user of the apparatus before the apparatus reproduces any of the AV data, comprising:

an AV buffer ~~which buffers~~ to buffer the AV data;

an AV reproduction engine ~~which decodes~~ to decode the AV data;

an enhanced navigation (ENAV) buffer ~~which preloads~~ to preload the markup document before the apparatus reproduces any of the AV data to enable the apparatus to reproduce the AV data in the interactive mode selected by the user;

an ENAV engine ~~which identifies~~ to identify buffering state information of the markup document and ~~decodes~~ decode the markup document, the buffering state information being

used by the apparatus in reproducing the AV data in the interactive mode selected by the user; and

~~means for obtaining~~ an I/O manager to obtain the markup document.

24. (Currently amended) The apparatus of claim 23, wherein the ~~apparatus~~ I/O manager uses a blocked I/O method ~~in response to obtaining~~ obtain the markup document from a data storage medium, and uses an unblocked I/O method ~~in response to obtaining~~ obtain the markup document from a network.

25. (Currently amended) The apparatus of claim 1, further comprising a reader ~~which reads~~ to read a preload-list file before the reproducing of the AV data begins in the interactive mode selected by the user;

wherein the buffer manager manages the buffer to preload the markup document based on contents of the preload-list file before the reproducing of the AV data begins in the interactive mode selected by the user.

26. (Currently amended) The apparatus of claim 25, wherein the preload-list file contains information identifying at least one markup document that is to be preloaded into the buffer under ~~the~~ control of the buffer manager before the reproducing of the AV data begins in the interactive mode selected by the user.

27. (Previously presented) The apparatus of claim 25, wherein the reader reads the preload-list file from an information storage medium.

28. (New) The apparatus of claim 1, wherein:

the interactive mode is a mode in which the AV data is displayed in a display window defined by the markup document;

the apparatus is selectively operable in the interactive mode in which the AV data is displayed in the display window defined by the markup document, and a non-interactive video mode in which the AV data is displayed in the same manner as AV data recorded on a standard DVD; and

the user of the apparatus selects between the interactive mode and the non-interactive video mode.